

# PREDIKSI PENYEBARAN HAMA PENGGEREK BATANG DI KABUPATEN BANDUNG BERDASARKAN INFORMASI CUACA DENGAN MENGGUNAKAN ALGORITMA ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM (ANFIS)

## *Prediction of Spread the Drill Stem Pest Based on Weather Information Using Adaptive Neuro-Fuzzy Inference System Algorithm in Kabupaten Bandung*

Mayriskha Isna Indriyani<sup>1</sup>, Fhira Nhita S.T., M.T.<sup>2</sup>, Dr. Deni Saepudin S.Si., M.Si.<sup>3</sup>

<sup>1,2,3</sup> Ilmu Komputasi, Fakultas Informatika, Universitas Telkom

<sup>1</sup>[mavriskhaii@gmail.com](mailto:mavriskhaii@gmail.com), <sup>2</sup>[farid.alchair@gmail.com](mailto:farid.alchair@gmail.com), <sup>3</sup>[denis7579@gmail.com](mailto:denis7579@gmail.com)

### ABSTRAK

Dalam tugas akhir ini yang berjudul "Prediksi Penyebaran Hama Penggerék Batang Berdasarkan Informasi Cuaca di Kabupaten Bandung dengan Menggunakan Algoritma *Adaptive Neuro Fuzzy Inference System* (ANFIS)" menggunakan data penyebaran hama yang didapatkan dari Dinas Pertanian Soreang. Karena data yang didapat banyak mengandung *missing value*, maka dilakukan penanganan *missing value* dengan menggunakan metode interpolasi linier. Data akan dipartisi menjadi 2 bagian, yaitu data *training* dan data *testing* dengan menggunakan *crossvalidation*.

Setelah dilakukan penelitian dari skenario dengan penggunaan PCA pada rata-rata keseluruhan kecamatan didapatkan hasil performansi WMAPE untuk *training* sebesar 0,23% dan untuk *testing* sebesar 134,99% sedangkan skenario tanpa penggunaan PCA didapatkan hasil performansi WMAPE untuk *training* sebesar 0,10% dan untuk *testing* sebesar 116,30%. Dari hasil performansi WMAPE yang dihasilkan dapat disimpulkan bahwa penelitian ini belum menghasilkan performansi yang cukup akurat untuk sebuah prediksi, hal ini dikarenakan data yang digunakan mengandung banyak *missing value* dan record yang terlalu sedikit.

**Kata Kunci :** *Adaptive Neuro-Fuzzy Inference System*, penyebaran hama, prediksi, *crossvalidation*, PCA.

### ABSTRACT

In this final task, entitled "Prediction of Spread the Drill Stem Pest Based on Weather Information Using Adaptive Neuro-Fuzzy Inference System Algorithm in Kabupaten Bandung" using pest distribution data obtained from the Department of Agriculture Soreang. Because the data obtained contains many missing value, then the handling missing value by using a linear interpolation method. Data will be partitioned into two parts, namely the training data and the data of testing using crossvalidation.

After a study of scenarios with the use of PCA on overall average performance results obtained districts for the training error of 0.23% and for the testing of 134,99% while the scenario without the use of PCA showed error performance for training by 0,10% and testing amounted to 116,30%. From the results of the performance of the resulting error can be concluded that this research has not produced enough performance for an accurate prediction.

**Keywords:** *Adaptive Neuro-Fuzzy Inference System*, the spread of pests, prediction, *crossvalidation*, PCA.

## 1. Pendahuluan

### 1.1 Latar Belakang

Indonesia terletak diantara 95° BT - 141° BT dan 6° LU- 11° LS dengan letak astronomis tersebut, Indonesia termasuk kedalam negara Agraris, yaitu negara yang penduduknya sebagian besar bekerja pada sektor pertanian. Kondisi pertanian di Indonesia saat ini sudah dalam keadaan yang baik, terbukti dengan banyaknya hasil pertanian yang diekspor ke luar negeri. Namun peningkatan produksi pertanian seringkali dihadapkan pada permasalahan gangguan serangan hama. Penggerék batang merupakan salah satu hama utama pada tanaman padi yang sering menimbulkan kerusakan dan kehilangan hasil secara nyata bahkan dapat mengakibatkan gagal

panen [1]. Perkembangan hama dipengaruhi oleh faktor-faktor iklim baik langsung maupun tidak langsung. Temperatur, kelembaban udara relatif dan fotoperiodisitas berpengaruh langsung terhadap siklus hidup, lama hidup, serta kemampuan diapause serangga [2].

Gangguan serangan hama penggerek batang pada tanaman padi sangat merugikan, sehingga pengendaliannya harus senantiasa diupayakan. Oleh sebab itu dalam tugas akhir ini dibangun sebuah arsitektur yang dapat memprediksi serangan hama penggerek batang pada tanaman padi. *Soft Computing* merupakan salah satu tools yang dapat digunakan untuk memprediksi serangan hama pada tanaman.

*Soft Computing* merupakan sekumpulan metodologi yang berkembang secara terus-menerus yang tujuan utamanya adalah untuk menghasilkan mesin ber-IQ tinggi. Teknik-teknik dasar yang termasuk ke dalam *Soft Computing* adalah *Fuzzy Logic*, *Neuro Computing* atau *Artificial Neural Networks*, *Evolutionary Computation*, *Probabilistic Computing*, dan teknik-teknik lain yang serupa. Terdapat banyak kombinasi yang bisa dibangun dari teknik-teknik dasar *Soft Computing*, empat diantaranya adalah *Neuro-fuzzy*, *Evolving ANN*, *Evolving Fuzzy*, *Fuzzy EAs* [3]. Dalam tugas akhir ini, *hybrid algorithm* yang digunakan adalah *Adaptive Neuro-Fuzzy Inference System* atau yang sering disebut dengan ANFIS.

Algoritma Adaptive Neuro Fuzzy Inference System (ANFIS) dipilih karena Jaringan Syaraf Tiruan memiliki kemampuan untuk *learning* dengan berdasarkan *training* pada data yang ada sebelumnya dan Logika Fuzzy mampu melakukan klasifikasi berdasarkan informasi linguistik. Kombinasi ini diharapkan dapat menghasilkan suatu sistem yang mampu *learning* secara terus menerus dan mampu memberikan hasil prediksi yang memiliki performansi *error* yang kecil. Penelitian sebelumnya yang berjudul *Adaptive Neuro-Fuzzy Inference Systems for Epidemiological Analysis Of Soybean Rust* memiliki akurasi sebesar 85%.

Pada penelitian tugas akhir ini dilakukan dengan prediksi time series. Target keluaran penelitian tugas akhir yaitu berupa prediksi penyebaran hama penggerek batang yang diharapkan dapat membantu petani untuk mengendalikan serangan hama penggerek batang pada tanaman padi agar hasil produksi tidak mengalami penurunan.

## 1.2 Perumusan Masalah

Rumusan masalah dari tugas akhir ini sebagai berikut:

1. Bagaimana memprediksi penyebaran hama penggerek batang pada tanaman padi berdasarkan informasi cuaca dan informasi luas serangan dengan mengimplementasikan algoritma ANFIS?
2. Bagaimana performansi algoritma ANFIS untuk memprediksi penyebaran hama penggerek batang pada tanaman padi berdasarkan informasi cuaca dan informasi luas serangan?

Adapun batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. Menggunakan data bulanan luas penyebaran hama pada tanaman padi khususnya hama penggerek batang di wilayah Kabupaten Bandung selama 4 tahun (2009-2012).
2. Menggunakan data cuaca bulanan di wilayah Kabupaten Bandung selama 4 tahun (2009-2012).
3. Data cuaca yang digunakan hanya data curah hujan dan semua kecamatan curah hujan dianggap sama jika di bulan yang sama.
4. Menggunakan 3 nilai linguistik (rendah, sedang, tinggi), 3 cluster FCM, dan fungsi keanggotaan *gaussianbell*.

## 1.3 Tujuan

Dari beberapa permasalahan yang ada maka tujuan yang ingin dicapai dari tugas akhir ini adalah sebagai berikut:

1. Mengetahui bagaimana memprediksi penyebaran hama penggerek batang pada tanaman padi berdasarkan informasi cuaca dan informasi luas serangan dengan mengimplementasikan algoritma ANFIS.
2. Mengetahui bagaimana performansi algoritma ANFIS untuk memprediksi penyebaran hama penggerek batang pada tanaman padi berdasarkan informasi cuaca dan informasi luas serangan.

## 2. Kajian Pustaka

### 2.1 Hama dan Tanaman

Hama dan penyakit tanaman adalah organisme yang mengganggu tanaman budidaya sehingga pertumbuhan dan perkembangan tanaman budidaya terhambat [4]. Yang dimaksud dengan hama adalah semua binatang yang merugikan tanaman, terutama yang berguna dan dibudidayakan manusia; apabila tidak merugikan tanaman yang berguna dan dibudidayakan manusia dengan sendirinya tidak disebut sebagai hama [5].

### 2.1.1 Tanaman Padi

Tanaman padi merupakan tanaman semusim termasuk golongan rumput-rumputan. Padi selain merupakan tanaman termuda yaitu tanaman yang biasanya berumur pendek, kurang dari satu tahun dan hanya satu kali berproduksi, setelah berproduksi akan mati atau dimatikan. Padi dapat hidup di tanah kering atau basah. Agar produksi padi maksimal maka padi harus ditanam pada lahan yang subur [7]. Padi dapat tumbuh baik di lingkungan yang panas dan kadar kelembaban tinggi dengan curah hujan rata-rata 200 mm perbulan dengan distribusi selama 4 bulan [8]. Suhu yang baik untuk tanaman ini yaitu 23°C dan dengan ketinggian tempat antara 0-1500 meter diatas permukaan laut (dpl) [8].

### 2.1.2 Penggerek Batang Padi

Penggerek batang termasuk hama paling penting pada tanaman padi yang sering menimbulkan kerusakan berat dan kehilangan hasil yang tinggi. Dilapang, keberadaan hama ini ditandai oleh kehadiran ngengat(kupu-kupu), kematian tunas-tunas padi (sundep, *dead heart*), kematian malai (beluk, *white head*), dan ulat (*larva*) penggerek batang.

Hama ini dapat merusak tanaman pada semua fase tumbuh, baik pada saat di pembibitan, fase anakan, maupun fase berbunga. Bila serangan terjadi pada pembibitan sampai fase anakan, hama ini disebut sundep dan jika terjadi pada saat berbunga, disebut beluk.

## 2.2 Interpolasi

Metode Interpolasi adalah suatu cara untuk mencari nilai di antara beberapa titik data yang telah diketahui. Tujuan dari interpolasi adalah untuk menentukan titik-titik antara dari n buah titik dengan menggunakan suatu fungsi pendekatan tertentu. Metode Interpolasi terdiri dari :

### - Interpolasi Linier

Interpolasi Linier adalah metode interpolasi untuk menentukan titik-titik antara dari 2 buah titik dengan menggunakan garis lurus.

Persamaan garis lurus yang melalui 2 titik  $P_1(x_1, y_1)$  dan  $P_2(x_2, y_2)$  dapat dituliskan dengan:

(2.1)

Sehingga diperoleh persamaan dari interpolasi linier sebagai berikut:

$$\left( \frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1} \right)$$

(2.2)

Algoritma Interpolasi Linier :

- 1) Tentukan dua titik  $P_1$  dan  $P_2$  dengan koordinatnya masing-masing  $(x_1, y_1)$  dan  $(x_2, y_2)$
- 2) Tentukan nilai  $x$  dari titik yang akan dicari
- 3) Hitung nilai  $y$  dengan :

$$\left( \frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1} \right)$$

(2.2)

- 4) Tampilkan nilai titik yang baru  $Q(x, y)$

## 2.3 Principal Component Analysis (PCA)

PCA adalah sebuah teknik untuk membangun variable-variable baru yang merupakan kombinasi linear dari variable-variable asli. PCA adalah pengolahan data yang populer dan merupakan salah satu teknik reduksi dimensi untuk tipe data numerik. Pada tugas akhir ini data memiliki satuan yaitu Ha, sehingga metode PCA yang digunakan adalah metode kovariansi.

### 2.3.1 Metode kovariansi

Pada referensi [10] dijelaskan proses kerja PCA dengan menggunakan metode kovariansi dengan langkah-langkah sebagai berikut :

1. Mengurangi setiap record dari data asli  $X$  dengan *mean*  $\bar{X}$  sehingga menghasilkan data hasil standarisasi yaitu *DataAdjust* dengan *mean* = 0.
2. Menentukan matriks kovariansi  $S$  dari *DataAdjust*.
3. Menghitung nilai eigen  $L$  dan vektor eigen  $U$  dari matriks  $S$ . Vektor eigen bersifat *orthonormal* sehingga :

(2.4)

4. Mengurutkan nilai eigen secara terurut menurun. Hubungan nilai eigen, vektor eigen, dan matriks

kovariansi dinyatakan dengan persamaan :

(2.5)

5. Mengubah  $p$  dimensi yang berkorelasi  $x_1, x_2, \dots, x_p$  menjadi  $p$  dimensi baru yang tidak berkorelasi  $z_1, z_2, \dots, z_p$ , dengan persamaan :

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1p} \\ a_{21} & a_{22} & \dots & a_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1} & a_{p2} & \dots & a_{pp} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix} \quad (2.6)$$

6. Dimensi baru yang dihasilkan dari kombinasi linier dimensi asli tersebut dinamakan *principal component* (PC) dari  $X$ . Setiap PC memiliki  $mean=0$  dan  $variansi=1$ . Nilai PC pada setiap record disebut *z-scores* yang dinyatakan dengan persamaan :

$$PC \text{ ke-} i = \frac{z_i - \bar{z}_i}{s_i} \quad (2.7)$$

## 2.4 K-Fold Cross Validation

Pengukuran kinerja model pada set data uji sangat penting untuk dilakukan, karena ukuran yang didapatkan memberikan perkiraan tidak bisa dari generalisasi error. Akurasi atau error yang dihitung dari set data uji juga bisa digunakan untuk membandingkan kinerja relatif dari klasifikator berbeda pada domain yang sama. Salah satu metode pengujian yang dapat digunakan untuk mengukur kinerja klasifikator adalah *K-Fold Cross Validation*. Pendekatan ini memecah set data menjadi  $k$  bagian set data dengan ukuran yang sama. Setiap kali berjalan, satu pecahan berperan sebagai set data latih sedangkan pecahan lainnya menjadi set data uji. Prosedur tersebut dilakukan sebanyak  $k$  kali sehingga setiap data berkesempatan menjadi data uji tepat satu kali dan menjadi data latih sebanyak  $k-1$  kali [12].

## 2.5 Fuzzy C-Means

*Fuzzy C-Means* adalah suatu teknik pengelompokan data yang keberadaan tiap data dalam suatu cluster ditentukan oleh derajat keanggotaannya. Teknik ini pertama kali dikemukakan oleh Dunn (1973) dan dikembangkan oleh Jim Bezdek pada tahun 1981 [13].

Algoritma Fuzzy C-Means secara lengkap dijelaskan secara berikut [14]:

1. Tentukan:
  - Matriks  $X$  berukuran  $n \times m$ , dengan  $n$  = jumlah data yang akan dicluster dan  $m$  = jumlah kriteria (variabel).
  - Jumlah cluster yang akan dibentuk.
  - Pangkat (bobot  $w > 1$ ).
  - Maksimum iterasi.
  - Kriteria penghentian ( $\epsilon$  = nilai positif yang kecil).
2. Bentuk matriks partisi awal  $U$  (derajat keanggotaan dalam cluster), matriks partisi awal dibuat secara acak:

$$U = \begin{bmatrix} ( ) & ( ) & ( ) \\ ( ) & ( ) & ( ) \\ ( ) & ( ) & ( ) \end{bmatrix} \quad (1)$$

3. Hitung pusat cluster  $V$  untuk setiap cluster:

$$V_j = \frac{\sum_{i=1}^n (u_{ij})^w x_i}{\sum_{i=1}^n (u_{ij})^w} \quad (2)$$

4. Hitung nilai objektif dengan rumus :

$$J = \sum_{i=1}^n \sum_{j=1}^c (u_{ij})^w \|x_i - V_j\|^2 \quad (3)$$

5. Perbaiki derajat keanggotaan setiap data pada tiap cluster (perbaiki matriks partisi):

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left( \frac{\|x_i - V_j\|}{\|x_i - V_k\|} \right)^{\frac{2}{w-1}}} \quad (4)$$

dengan:

$$\|x_i - V_j\| = \sqrt{\sum_{k=1}^m (x_{ik} - V_{jk})^2} \quad (5)$$

6. Tentukan kriteria penghentian iterasi, yaitu perubahan matriks partisi:

Apabila  $\| \mathbf{e} \| < \epsilon$  maka iterasi dihentikan.

7. Jika iterasi berhenti, ditentukan cluster dari tiap-tiap data. Cluster dipilih berdasarkan nilai matriks partisi terbesar.

## 2.6 Adaptive Neuro Fuzzy Inference System (ANFIS)

Adaptive Neuro Fuzzy Inference System (ANFIS) adalah arsitektur yang secara fungsional sama dengan fuzzy model sugeno. Arsitektur ANFIS juga sama dengan jaringan syaraf tiruan dengan fungsi radial dengan sedikit batasan tertentu. Bisa dikatakan ANFIS adalah suatu metode yang mana dalam melakukan penyetelan aturan-aturan digunakan algoritma pembelajaran terhadap sekumpulan data. ANFIS juga memungkinkan aturan-aturan untuk beradaptasi [15]. ANFIS (*Adaptive Neuro Fuzzy Inference System*) juga merupakan algoritma optimasi *Fuzzy Inference System* dengan menggunakan ANN.

Pada model Sugeno orde satu, himpunan aturan menggunakan kombinasi linier dari input-input yang ada dapat diekspresikan sebagai

$$\text{IF } x \text{ is } A_1 \text{ AND } y \text{ is } B_1 \text{ THEN } f_1 = p_1x + q_1y + r_1 \quad (2.21)$$

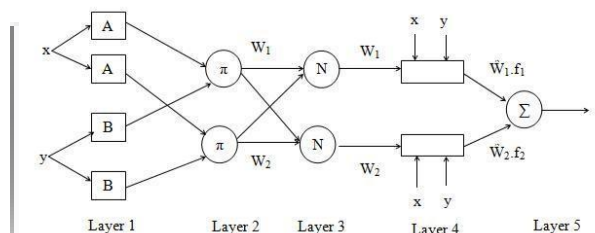
$$\text{IF } x \text{ is } A_2 \text{ AND } y \text{ is } B_2 \text{ THEN } f_2 = p_2x + q_2y + r_2 \quad (2.22)$$

Mekanisme penalaran pada model ini adalah

(2.23)

### 2.6.1 Arsitektur ANFIS

Selanjutnya, arsitektur ANFIS untuk kasus dua input, x dan y, serta satu output yang dilambangkan z diilustrasikan oleh gambar di bawah ini [3],



Gambar 2.6 Arsitektur ANFIS [3].

Berdasarkan referensi [3], arsitektur ANFIS terdiri dari 5 layer yang masing-masing layer memiliki fungsi-fungsi yang dapat dijelaskan sebagai berikut:

**Layer 1 :** Berfungsi sebagai proses *fuzzyfication*. Output dari node 1 pada layer 1 dinotasikan sebagai . Setiap node pada layer I bersifat *adaptive* dengan output :

$$(\quad) \quad (2.24)$$

$$(\quad) \quad (2.25)$$

Dimana x dan y adalah nilai-nilai input untuk node tersebut dan dan adalah himpunan *fuzzy*. Jadi, masing-masing node pada layer 1 berfungsi membangkitkan derajat keanggotaan (bagian *premise*).

Sebagai contoh, misalkan fungsi keanggotaan yang digunakan adalah *generalized bell (gbell)*, rumusnya adalah sebagai berikut:

$$(\quad) = \frac{1}{1 + \exp\left(-\frac{(x-a)^b}{c}\right)} \quad (2.26)$$

dimana {a,b,c} adalah parameter-parameter.

**Layer 2 :** Tiap-tiap neuron pada lapisan ke dua berupa neuron tetap yang outputnya adalah hasil dari masukan. Biasanya digunakan operator AND. Pada layer 2 dinotasikan  $\pi$ , Setiap node pada layer ini berfungsi untuk menghitung kekuatan aktivasi (*firing strength*) pada setiap *rule* sebagai *product* dari semua *input* yang masuk atau sebagai operator *t-norm (triangular norm)*:

$$( ) \quad ( ) \quad (2.27)$$

Sehingga,

$$( ) \quad ( ) \quad (2.28)$$

$$( ) \quad ( ) \quad (2.29)$$

**Layer 3 :** Dilambangkan dengan N. Setiap node pada lapisan ini bersifat *non-adaptif* yang berfungsi hanya menghitung rasio antara *firing strength* pada rule ke-i terhadap total *firing strength* dari semua *rule* :

$$(2.30) \text{---}$$

**Layer 4 :** Setiap node pada lapisan ini bersifat adaptif sebagai fungsi

$$( ) \quad ( ) \quad \text{dengan } i=1,2 \quad (2.31)$$

Dimana adalah output dari layer 3 dan ( ) adalah himpunan parameter pada *fuzzy* model Sugeno orde pertama. Parameter-parameter pada lapisan ini disebut dengan *consequent parameters*.

**Layer 5 :** Pada layer ini , tiap-tiap neuron adalah node tetap yang merupakan jumlahan dari semua masukan. Satu mode tunggal yang dilambangkan  $\Sigma$  pada layer ini berfungsi mengagregasikan seluruh *output* dari layer 4 (yang didefinisikan sebagai penjumlahan dari semua sinyal yang masuk) :

$$\Sigma = \frac{\Sigma \text{---}}{\Sigma \text{---}} \quad (2.32)$$

Dengan demikian, kelima *layer* tersebut akan membangun suatu *adaptive-network* yang secara fungsional ekuivalen dengan *fuzzy* model Sugeno orde satu [3].

Perubahan nilai parameter , adalah sebagai berikut :

$$\text{dan} \quad (2.56)$$

$$(2.57)$$

Dengan adalah laju pembelajaran yang terletak pada interval [0,1]. Sehingga dihasilkan parameter nilai dan nilai yang baru, sebagai berikut :

$$( ) \quad \text{dan} \quad (2.58)$$

$$( ) \quad (2.59)$$

## 2.7 Performansi Error

WMAPE adalah salah satu pengukur kesalahan dalam kasus prediksi. WMAPE dihitung menggunakan rata-rata kesalahan absolut dibagi dengan rata-rata data aktual. Rumus WMAPE sebagai berikut [17] :

$$\Sigma \text{---} \quad (2.60)$$

Dimana:

$$\Sigma ( )$$

= Nilai aktual ke-i

= Nilai prediksi ke-i

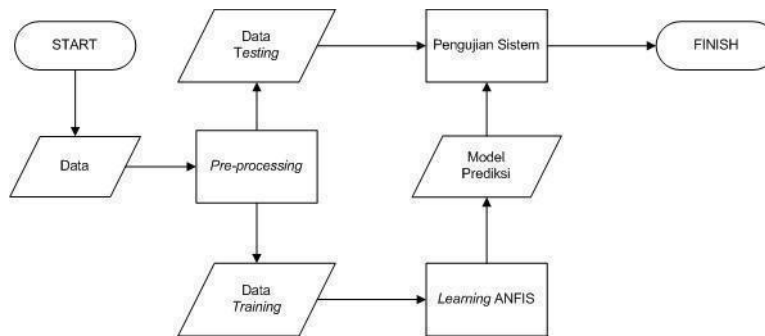
n = Banyaknya data

## 3 Metodologi dan Desain Sistem

### 3.1 Deskripsi Sistem



Pada tugas akhir ini akan dirancang suatu sistem untuk memprediksi penyebaran hama penggerek batang satu bulan kedepan di Kabupaten Bandung menggunakan data penyebaran hama bulanan dan data curah hujan bulanan dari tahun 2009-2012. Berikut gambaran umum sistem :



**Gambar 1.** Gambaran Umum Sistem

Proses yang dilakukan pada gambar diatas dimulai dari proses *pre-processing*, didalam tahapan *pre-processing* terdapat tahapan interpolasi untuk menangani *missing value*, PCA untuk mereduksi data, dan normalisasi untuk mengubah data menjadi *range* antara 0 sampai 1. Selanjutnya data hasil normalisasi dipartisi menjadi data *training* dan *testing* dengan menggunakan *crossvalidasi*. Sebelum masuk ke pelatihan ANFIS, data data *training* dikelompokkan menggunakan *fuzzy c-means* untuk menghasilkan nilai standar deviasi dan mean untuk digunakan sebagai parameter inputan awal di layer 1 pada ANFIS. Fungsi keanggotaan yang digunakan adalah Fungsi *GaussianBell*. Setelah itu dilakukan proses ANFIS. Selanjutnya, hitung performansi WMAPE untuk data *training* sehingga mendapatkan parameter yang optimal. Parameter optimal yang didapatkan dari proses training dilakukan pengujian menggunakan data *testing*. Hasil dari penelitian ini dengan menggunakan algoritma ANFIS adalah berupa nilai prediksi yang nantinya diharapkan akan memberikan performansi WMAPE yang akurat.

### 3.2 Dataset

Dataset yang digunakan pada penelitian ini adalah data penyebaran hama dan data curah hujan di Kabupaten Bandung dari 2009-2012.

Tabel 3.1. Sampel data penyebaran hama

| Periode    | Ciwidey | Pasirjambu | Rancabali | Cimaung |  | Cilengkrang | Cimenyan |
|------------|---------|------------|-----------|---------|--|-------------|----------|
| Feb 2009   | 0       | 0          | 0         | 0       |  | 4,6         | 3,6      |
| Mar 2009   | 0       | 0          | 0         | 0       |  | 3           | 2        |
| April 2009 | 0       | 0          | 0         | 0       |  | 1,4         | 0,4      |
| Mei 2009   | 0       | 0          | 0         | 0       |  | 0           | 2,75     |
| Juni 2009  | 0       | 0          | 0         | 0       |  | 0,5         | 1,5      |
| Juli 2009  | 0       | 0          | 0         | 0       |  | 3           | 1        |
| Agust 2009 | 0       | 0          | 0         | 0       |  | 1,5         | 2        |
|            |         |            |           |         |  |             |          |
| Juli 2012  | 21      | 0          | 0         | 0       |  | 3.5         | 1        |

### 3.3 Skenario Pengujian

Skenario yang akan digunakan untuk pelatihan dan pengujian pada tugas akhir ini yang bertujuan untuk mendapatkan arsitektur terbaik dengan *error* terkecil adalah sebagai berikut

1. Tanpa menggunakan metode PCA
  - a. Membagi proposi data dengan menggunakan *crossvalidation*
  - b. Jumlah *input* yang digunakan adalah 31 *input*. 31 *input* terdiri dari 30 atribut kecamatan dan 1 atribut curah hujan.

- c. Jumlah nilai *linguistic* yang digunakan adalah 3, terdiri dari rendah, sedang, dan tinggi
  - d. Fungsi keanggotaan yang digunakan adalah fungsi keanggotaan *gaussianBell*.
  - e. Learning rate yang digunakan adalah 0.001, 0.01, dan 0.1.
  - f. Epoch yang digunakan dalam proses pelatihan adalah 200.
2. Menggunakan metode PCA
    - a. Membagi proposi data dengan menggunakan *crossvalidation*
    - b. Jumlah *input* yang digunakan dari hasil reduksi data adalah
      - 9 *input* terdiri dari 8 atribut hasil PCA dan 1 atribut curah hujan
      - 10 *input* terdiri dari 9 atribut hasil PCA dan 1 atribut curah hujan
      - 11 *input* terdiri dari 10 atribut hasil PCA dan 1 atribut curah hujan
    - c. Jumlah nilai *linguistic* yang digunakan adalah 3, terdiri dari rendah, sedang, dan tinggi
    - d. Fungsi keanggotaan yang digunakan adalah fungsi keanggotaan *gaussianBell*.
    - e. Learning rate yang digunakan adalah 0.001, 0.01, dan 0.1.
    - f. Epoch yang digunakan dalam proses pelatihan adalah 200.

#### 4. Pengujian dan Analisis

##### 4.1 Hasil Pengujian dan Analisis Penggunaan PCA

##### 4.1.1 Analisis Penggunaan PCA

Berikut merupakan hasil *training* dan *testing* menggunakan data hasil dari reduksi dengan menggunakan PCA, sebagai berikut

Tabel 4.1 Hasil WMAPE *training* dan *testing* dengan PCA

| Learning Rate | Atribut | Training |       |       |              | Testing |         |         |                |
|---------------|---------|----------|-------|-------|--------------|---------|---------|---------|----------------|
|               |         | 1        | 2     | 3     | Rata-Rata    | 1       | 2       | 3       | Rata-Rata      |
| 0,001         | 11      | 0,60%    | 0,37% | 0,82% | <b>0,60%</b> | 178,68% | 175,28% | 169,74% | <b>174,57%</b> |
|               | 10      | 0,84%    | 0,60% | 1,09% | <b>0,84%</b> | 180,82% | 187,00% | 188,31% | <b>185,37%</b> |
|               | 9       | 8,22%    | 4,71% | 2,55% | <b>5,16%</b> | 199,82% | 215,56% | 176,44% | <b>197,27%</b> |
| 0,01          | 11      | 0,23%    | 0,28% | 0,18% | <b>0,23%</b> | 144,88% | 121,93% | 138,17% | <b>134,99%</b> |
|               | 10      | 0,78%    | 1,12% | 0,90% | <b>0,93%</b> | 152,86% | 194,24% | 165,74% | <b>170,95%</b> |
|               | 9       | 0,60%    | 0,30% | 1,72% | <b>0,87%</b> | 164,74% | 197,86% | 175,13% | <b>179,24%</b> |
| 0,1           | 11      | 0,45%    | 3,71% | 0,60% | <b>1,58%</b> | 198,66% | 128,06% | 164,93% | <b>163,88%</b> |
|               | 10      | 1,08%    | 1,25% | 3,27% | <b>1,87%</b> | 218,27% | 187,80% | 199,08% | <b>201,72%</b> |
|               | 9       | 7,76%    | 4,69% | 2,46% | <b>4,97%</b> | 167,75% | 820,69% | 172,99% | <b>387,14%</b> |

Dari Tabel 4.1 didapatkan nilai WMAPE terkecil terletak pada 11 atribut pada learning rate 0,01, dan dari keseluruhan terlihat bahwa nilai WMAPE yang diuji dengan menggunakan 11 atribut hasil PCA nilai WMAPE yang diperoleh lebih kecil dibandingkan dengan 10 atribut dan 9 atribut. Hal ini disebabkan karena nilai variansi dari hasil PCA dengan 11 atribut lebih besar jika dibandingkan dengan variansi hasil dari 10 atribut dan 9 atribut. Hasil variansi PCA dengan 11 atribut adalah 93.6563 sedangkan dengan menggunakan 10 atribut dan 9 atribut adalah 92.115 dan 90.2563, sehingga dapat disimpulkan semakin besar variansi PCA nya maka nilai WMAPE dari ANFIS akan semakin kecil.

Berdasarkan analisis penggunaan PCA didapatkan *error* terkecil adalah sebagai berikut

Tabel 4.2 Hasil *error* terkecil dengan PCA

| Learning Rate | WMAPE    | 11 atribut     |
|---------------|----------|----------------|
| 0,01          | Training | <b>0,23%</b>   |
|               | Testing  | <b>134,99%</b> |

##### 4.1.2 Analisis Tanpa Penggunaan PCA

Berikut merupakan hasil *training* dan *testing* tanpa menggunakan data hasil dari reduksi dengan menggunakan PCA, sebagai berikut

Tabel 4.3 Hasil *training* dan *testing* tanpa menggunakan PCA

| Learning Rate | Atribut | Training |       |       |           | Testing |         |         |           |
|---------------|---------|----------|-------|-------|-----------|---------|---------|---------|-----------|
|               |         | 1        | 2     | 3     | Rata-Rata | 1       | 2       | 3       | Rata-Rata |
| 0,001         | 31      | 0,09%    | 0,16% | 0,09% | 0,11%     | 130,58% | 152,75% | 116,28% | 133,20%   |
| 0,01          | 31      | 0,11%    | 0,09% | 0,09% | 0,10%     | 136,20% | 100,37% | 112,33% | 116,30%   |
| 0,1           | 31      | 0,19%    | 9,05% | 0,09% | 3,11%     | 156,90% | 142,39% | 128,01% | 142,44%   |

Dari Tabel 4.3 didapatkan nilai WMAPE testing terkecil sama seperti Tabel 4.1 yaitu terletak pada learning rate 0,01.

Tabel 4.4 Hasil *error* terkecil tanpa penggunaan PCA

| Learning Rate | WMAPE    | 31 atribut |
|---------------|----------|------------|
| 0,01          | Training | 0,10%      |
|               | Testing  | 116,30%    |

#### 4.1.3 Perbandingan Tanpa Penggunaan PCA dan Penggunaan PCA

Berikut merupakan hasil perbandingan antara penggunaan PCA dan tanpa penggunaan PCA, sebagai berikut

Tabel 4.5 Hasil Perbandingan nilai WMAPE antara penggunaan PCA dan tanpa penggunaan PCA

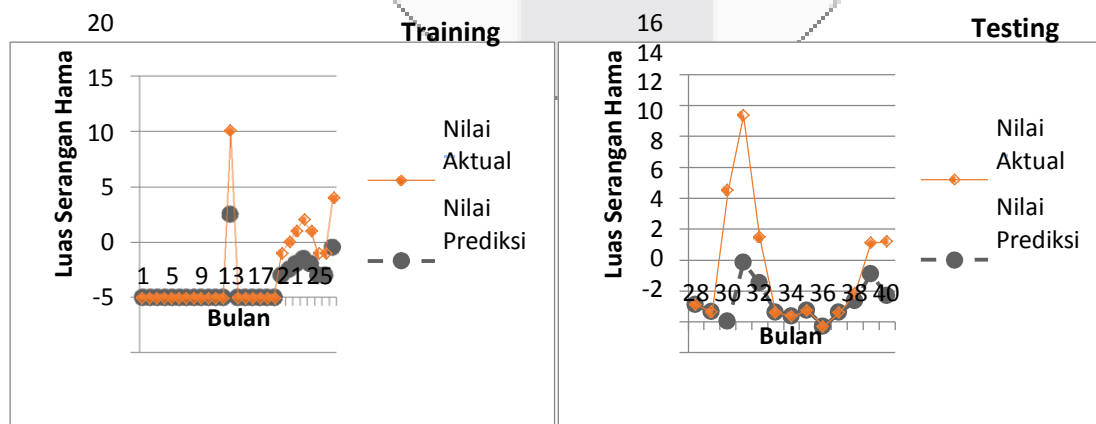
| Learning Rate | WMAPE    | 31 atribut | 11 atribut |
|---------------|----------|------------|------------|
| 0,01          | Training | 0,10%      | 0,23%      |
|               | Testing  | 116,30%    | 134,99%    |

Dari Tabel 4.5 terlihat bahwa tanpa penggunaan PCA nilai WMAPE yang di peroleh lebih kecil jika dibandingkan dengan penggunaan PCA. Hal ini disebabkan karena data yang menjadi inputan untuk ANFIS direduksi oleh PCA, sehingga data yang dihasilkan dari PCA jika diubah lagi ke data awal akan jauh berbeda dengan data asli nya.

Dari semua skenario yang digunakan, dihasilkan nilai WMAPE yang sangat besar, hal ini dikarenakan sistem tidak mampu memprediksi dengan baik. Dan setelah dilakukan pengamatan, kegagalan metode ANFIS dipengaruhi oleh data yang digunakan mengandung *missing value* yang sangat banyak dan dapat juga dipengaruhi oleh keterbatasan data atau record dataset yang bisa dibilang sangat sedikit untuk dijadikan sebuah inputan pada ANFIS.

Dari berbagai skenario didapatkan hasil performansi WMAPE terbaik untuk *training* sebesar 0,10% dan untuk *testing* sebesar 116,30% pada skenario tanpa penggunaan PCA.

Berdasarkan nilai WMAPE terbaik didapatkan arsitektur ANFIS terbaik untuk memprediksi. Berikut merupakan hasil prediksi per-kecamatan menggunakan arsitektur ANFIS terbaik.



Gambar 4.1 Hasil Prediksi di Arjasari

## 5. Kesimpulan dan Saran

### 5.1 Kesimpulan

Dari hasil pengujian dapat diambil beberapa kesimpulan, antara lain :

1. Dari hasil performansi WMAPE yang dihasilkan dapat disimpulkan bahwa penelitian ini belum menghasilkan performansi yang cukup akurat untuk sebuah prediksi, hal ini dikarenakan data yang digunakan mengandung banyak *missing value* dan *record* yang terlalu sedikit.
2. Berdasarkan skenario dengan penggunaan PCA pada rata-rata keseluruhan kecamatan didapatkan hasil performansi WMAPE untuk *training* sebesar 0,23% dan untuk *testing* sebesar 134,99% sedangkan skenario tanpa penggunaan PCA didapatkan hasil performansi WMAPE untuk *training* sebesar 0,10% dan untuk *testing* sebesar 116,30%.
3. Dari berbagai skenario didapatkan hasil performansi WMAPE terbaik untuk *training* sebesar 0,10% dan untuk *testing* sebesar 116,30% pada skenario tanpa penggunaan PCA.

### 5.2 Saran

Untuk mendapatkan performansi yang lebih baik pada perancangan berikutnya, terdapat beberapa saran yang dapat dijadikan sebagai bahan pertimbangan antara lain:

1. Diharapkan untuk menggunakan data yang memiliki *record* besar seperti lebih dari 100 *record* dan tidak mengandung banyak *missing value* agar nilai WMAPE yang dihasilkan lebih baik lagi.
2. Diharapkan parameter yang digunakan tidak hanya kecamatan dan curah hujan saja, karena banyak faktor yang mempengaruhi penyebaran hama seperti suhu, kecepatan angin, dan lain-lain.

### Daftar Pustaka

- [1] Webster, R.K. dan D.S. Mikkelsen. 1992. *Compendium of Rice Diseases*. APS Press. Minnesota
- [2] Susanti, Emi. 2008. Dampak Perubahan Iklim terhadap Serangan Organisme Pengganggu Tanaman (OPT) Serta Strategi Antisipasi dan Adaptasi. Balai Penelitian Agroklimate dan Hidrologi.
- [3] Suyanto, *Artificial Intelligence, Searching, Reasoning, Planning, and Learning*. Bandung: Informatika, 2007.
- [4] J.S, Ir. Dede Juanda., Cahyono, Ir Bambang. Ubi Jalar, Budidaya dan Analisis Usaha Tani. Yogyakarta: Kanisius, 2000.
- [5] Pracaya. Hama dan Penyakit Tanaman. Penebar Swadaya. Bogor, 1991.
- [6] Tjahjadi, Ir. Nur. Hama dan Penyakit Tanaman. Yogyakarta: Kanisius, 1989.
- [7] AAK. Budidaya Tanaman Padi. Aksi Agraris Kanisius. Yogyakarta, 1993.
- [8] Suprihatno, Bambang, dkk. Inovasi Teknologi Padi: Menuju Swasembada Beras Berkelanjutan. Bogor: Puslitbang Tanaman Pangan, 2005.
- [9] Mahyuni, dkk. Masalah Lapang Hama, Penyakit, Hara Pada Padi. Kerjasama BPTP Kalsel dan IRRI. Kalsel, 2010.
- [10] Fhira Nhita, "Analisis Principal Component Analysis (PCA) pada Unsupervised Learning untuk Data Berdimensi Tinggi". Laporan Tugas Akhir, Jurusan Teknik Informatika, STT Telkom.
- [11] G Instrument, 2010. "*Understanding Error and Accuracy*".
- [12] Prasetyo Eko. 2014. DATA MINING Mengolah Data Menjadi Informasi Menggunakan MATLAB. Yogyakarta
- [13] Sahedani Komal, "A Survey: Fuzzy set theory in Data mining", International Journal of Advanced Research in IT and Engineering, Vol.2, No.7, 2013
- [14] Krastev Georgi, Georgiev Tsvetozar, "Fuzzy Clustering Using C-Means Method", Bulgaria, Vol 4, No 2, 2015
- [15] Kusumadewi, Sri. Neuro-Fuzzy Integrasi Sistem Fuzzy & Jaringan Syaraf. Jogjakarta: Graha Ilmu, 2002.
- [16] Alves, Marcelo de Carvalho, dkk. 2011. *Adaptive Neuro-Fuzzy Inference System for Epidemiological Analysis of Soybean Rust*.
- [17] Wilson, Tom, 2012. "Forecast Accuracy and Uncertainty of Australian Bureau of Statistics State and Territory Population Projections". International Journal of Population Research